

07-280 Learning Objectives

Spring 2026 - Midterm 1

Course Level Learning Outcomes (to date)

1. Course Level
 - a. Implement and analyze existing algorithms, including well-studied methods for heuristic-based search, constraint satisfaction problems, classification, regression, and feature learning
 - b. Integrate multiple facets of practical AI and ML in a single system: problem formulation, data preprocessing, learning, regularization, and algorithm/model selection
 - c. Describe the formal properties of algorithms and models in AI and ML, and explain the practical implications of those results
 - d. Describe the trade-offs between computationally cheap, less efficient/accurate methods and computationally expensive, but more accurate methods, especially as they apply to heuristic implementation and optimization techniques
 - e. Design experiments to evaluate and compare different machine learning techniques on real-world problems
 - f. Employ probability, statistics, calculus, linear algebra, and optimization in order to develop new predictive models or learning methods
 - g. Given a description of an AI/ML technique, analyze it to identify (1) the expressive power of the formalism; (2) the size and complexity of the search space; (3) the computational properties of the algorithm; (4) any guarantees (or lack thereof) regarding termination, convergence, correctness, accuracy, or generalization power
 - h. Select and apply an appropriate supervised learning algorithm for classification and regression problems (e.g., decision trees, linear regression, logistic regression)

Search Fundamentals

1. Heuristic Search
 - a. Understand how to formulate and define problems as search problems.
 - b. Describe and implement various search algorithms.
 - c. Understand how search algorithms run on a given problem.
 - d. Analyze the time and space complexity of a variety of search algorithms.
 - e. Verify or create admissible/consistent heuristics for a given search problem
 - f. Describe the trade-offs between computationally cheap, inaccurate heuristics and computationally expensive heuristics
 - g. Understand both the algorithmic differences and the distinct use cases for tree search vs graph search
 - h. Discuss the ethical implications of a given real-world application involving search algorithms
2. Adversarial Search

- a. Formulate a real-world problem as a search problem over an adversarial search tree
 - b. Trace and implement algorithms for game trees with a mixture of node types (min/max/expectation) given a start state, player list, actions, and utilities/rewards
 - c. Describe the bounded lookahead algorithm
 - d. Define and create evaluation functions
 - e. Analyze a general game tree to determine which nodes can be pruned without causing an impact on the optimal solution
3. Constraint Satisfaction Problems
- a. Describe the definition of a CSP problem and its connection with general search problems
 - b. Formulate a real-world problem as a CSP
 - c. Describe and implement the backtracking algorithm
 - d. Define arc consistency
 - e. Describe and implement forward checking and AC-3
 - f. Explain the differences between MRV and LCV heuristics

ML Fundamentals

4. ML Overview
- a. Formulate a well-posed learning problem for a real-world task by identifying the task, performance measure, and training experience
 - b. Reason about the decision boundary for a classification algorithm (e.g., compare decision trees and logistic regression)
5. Decision Trees
- a. Formalize a learning problem by identifying the input space, output space, hypothesis, and hypothesis space
 - b. Implement decision tree training and prediction
 - c. Use effective splitting criteria for decision trees and be able to define entropy, conditional entropy, and mutual information
 - d. Judge whether a decision tree is underfitting or overfitting
 - e. Explain how early stopping method affects overfitting in decision tree learning
 - f. Describe the greedy nature of decision tree algorithms and the implications of achieving an optimal solution
 - g. Explain how decision trees could be used for regression tasks as well as classification tasks
6. Optimization for ML
- a. Identify the relationships between the concepts of loss, risk, empirical risk, and empirical risk minimization as they relate to finding a hypothesis function / model in machine learning
 - b. Given a loss function and hypothesis function structure (e.g., linear regression structure), formulate the machine learning optimization problem
 - c. Explain why there could be a need for more general loss functions than just zero-one loss for classification tasks
 - d. Demonstrate how squared error loss and mean squared error are related to empirical risk minimization for regression
 - e. Apply (batch) gradient descent, stochastic gradient descent, and mini-batch gradient descent to optimize an objective function

- f. Apply knowledge of zero derivatives to identify a closed-form solution (if one exists) to an optimization problem
 - g. Distinguish between convex and nonconvex functions and how they are related to various machine learning models
- 2. Calculus
 - a. Derive the gradient of a differentiable, multi-variable objective function by applying knowledge of multivariable calculus, including:
 - i. Partial derivatives involving scalars, vectors, and matrices
 - ii. Denominator layout of partial derivatives
 - b. Derive matrix calculus properties by expanding linear algebra into scalar form as needed
- 3. Linear Regression
 - a. Implement learning for linear regression using four optimization techniques: (1) closed form, (2) (batch) gradient descent, (3) stochastic gradient descent, and mini-batch stochastic gradient descent
 - b. Analyze the tradeoffs of computational complexity vs. convergence speed for linear regression optimization techniques on a particular dataset
 - c. Identify properties of datasets that affect the number of unique solutions to linear regression
- 4. Logistic Regression
 - a. Describe the differences between linear and logistic regression, including modeling, optimization, and application
 - b. Implement logistic regression for binary or multi-class classification
 - c. Explain the differences and similarities between binary and multi-class logistic regression models and loss functions
- 5. Model Selection
 - a. Plan an experiment that uses training, validation, and test datasets to predict the performance of a classifier on unseen data (without cheating)
 - b. Explain the difference between (1) training error, (2) cross-validation error, and (3) test error
 - c. For a given learning technique, identify the model, learning algorithm, parameters, and hyperparameters
 - d. Describe trade-offs between different options for cross-validation
- 6. Feature Engineering and Regularization
 - a. Engineer appropriate features for a new task
 - b. Convert a non-linear dataset (or linearly inseparable dataset) to a linear dataset (or linearly separable dataset) in higher dimensions
 - c. Explain why linear models are still considered linear despite their ability to be applied to tasks with highly non-linear datasets
 - d. Identify symptoms that indicate that a model is overfitting
 - e. Add a regularizer to an existing objective in order to combat overfitting

References

Built from 10-301/601 Learning objectives